Platform Conference

Direction • Design • Perspective • Analysis

Next Generation Packet Processing: RM9000x2™ Integrated Multiprocessor With HyperTransport™

Rob Reiner

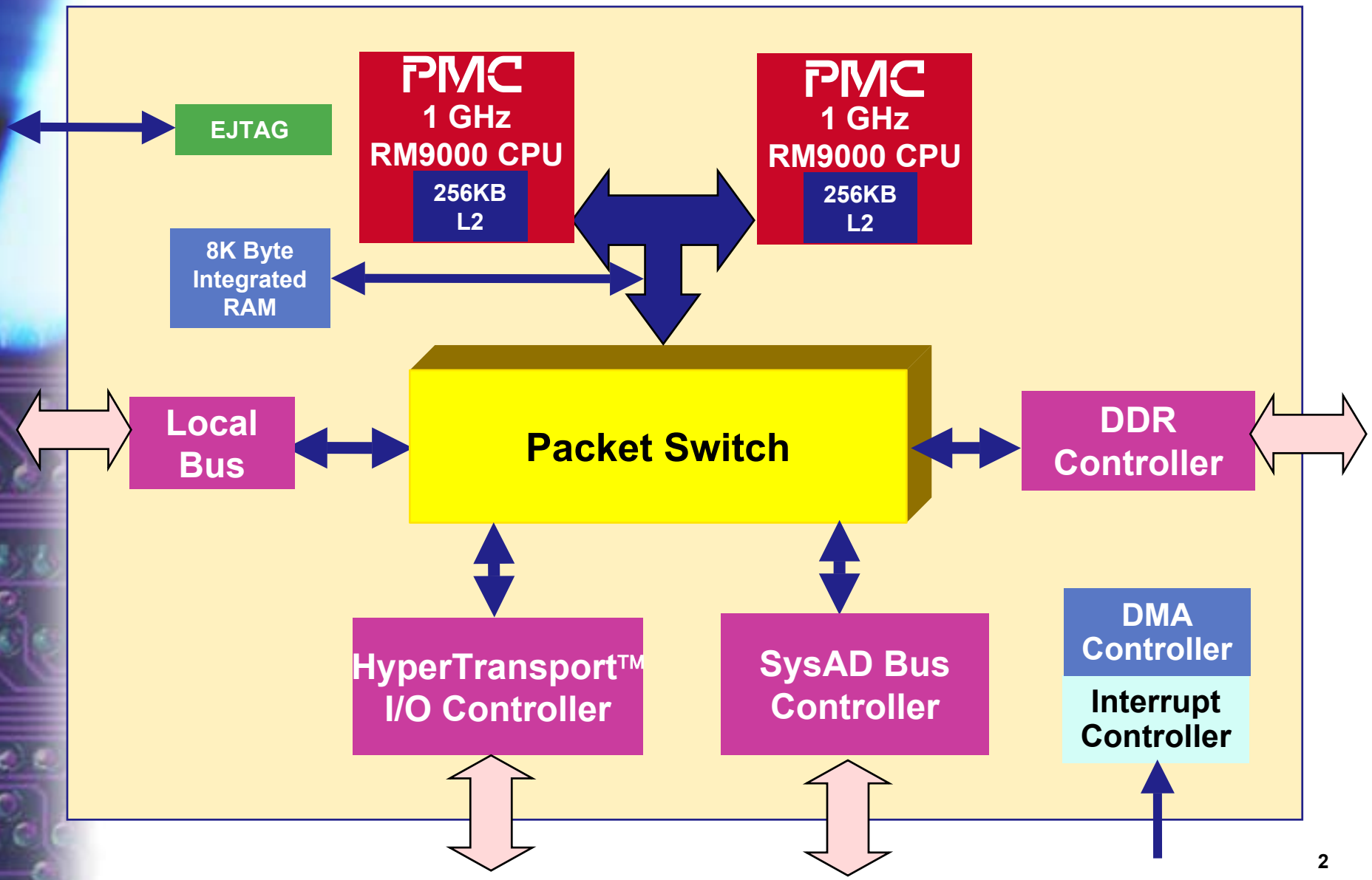
Product Marketing Manager

PMC-Sierra, Inc.

PMC-SIERRA

January 23-24, 2002

RM9000x2 Block Diagram



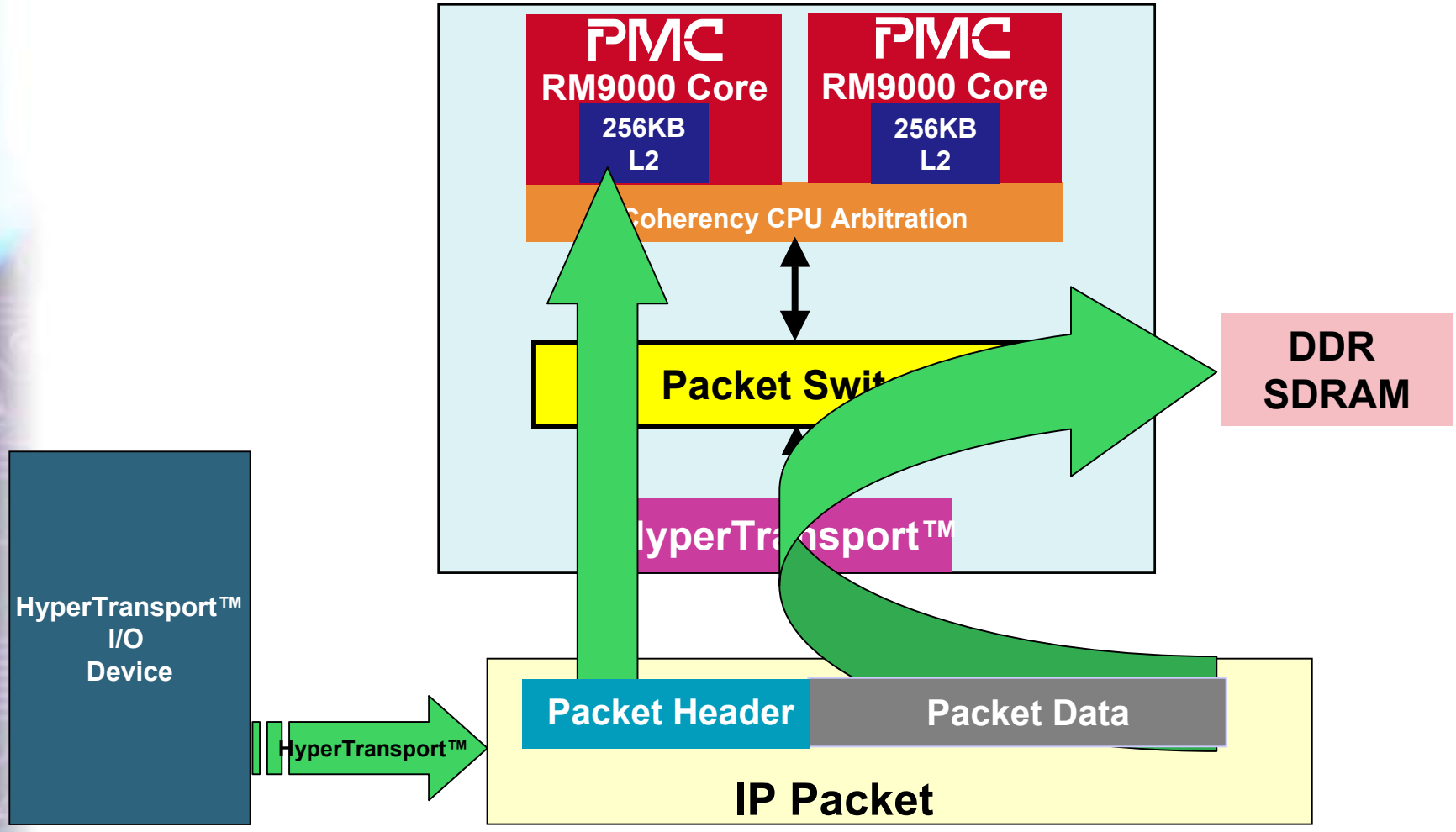
RM9000x2 Delivers Next Generation Packet Processing

- **Very High Performance I/O Subsystem**
  - 16 Gbit/s HyperTransport™ Interface
  - Direct Deposit Cache™ Permits I/O Devices Direct Access into Cache Memory
- **Very High Performance CPU Subsystem**
  - Dual GHz CPU Cores
  - Cache Coherency enforced between CPU Cores using 5-State MOESI Protocol

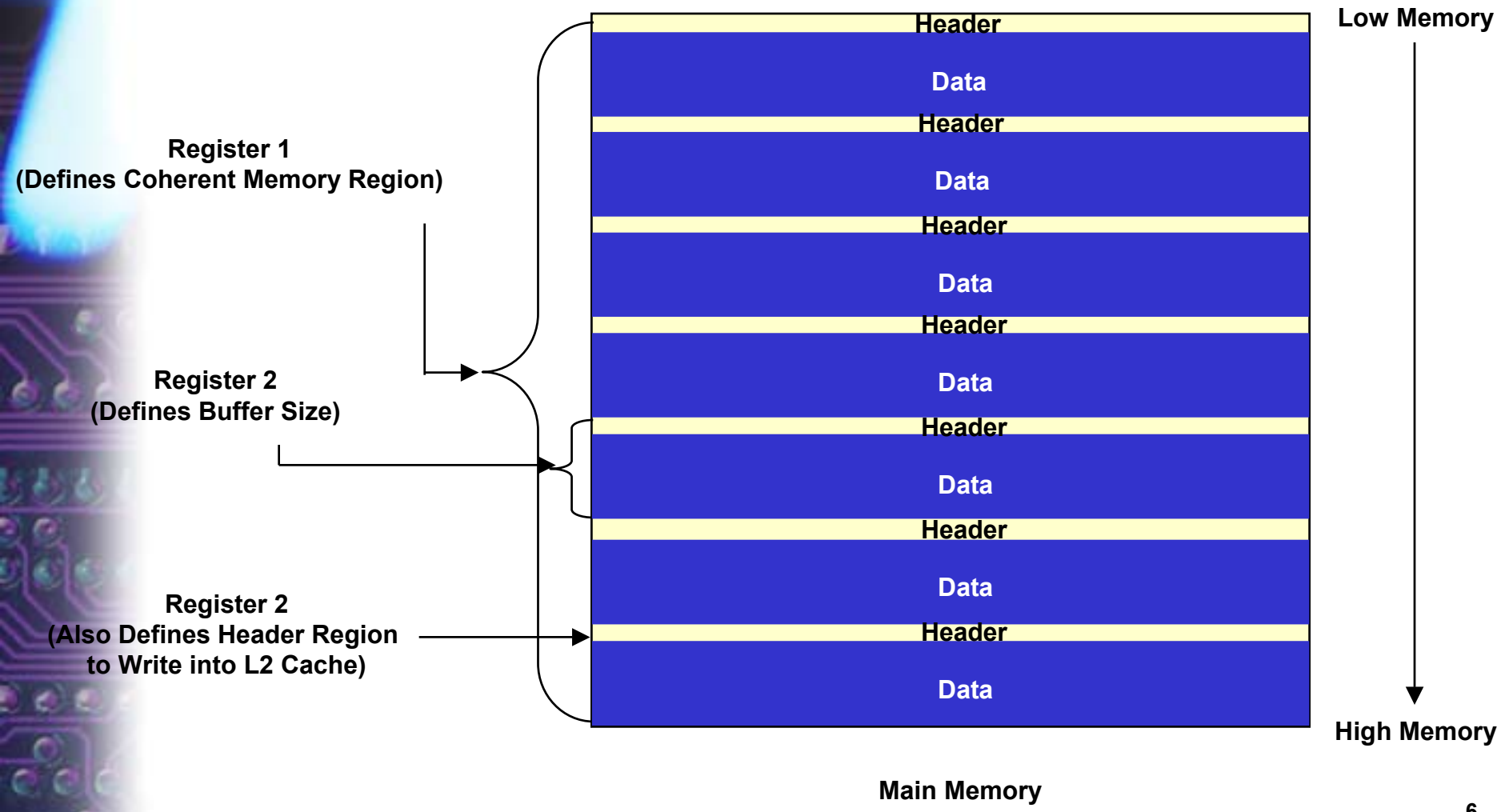
Direct Deposit Cache Provides 2 Methods for I/O to Access Cache

- **AutoDeposit™ Operation**
  - For All Packet Buffers Written over HyperTransport™ in Pre-defined Address Range
    - Packet Header Written to L2 Cache
    - Packet Data Written to Main Memory
- **Live Deposit™ Operation**
  - HyperTransport™ Write Command Header Decoded for Direct Cache Access
  - DMA Channels Provide Direct Cache Access

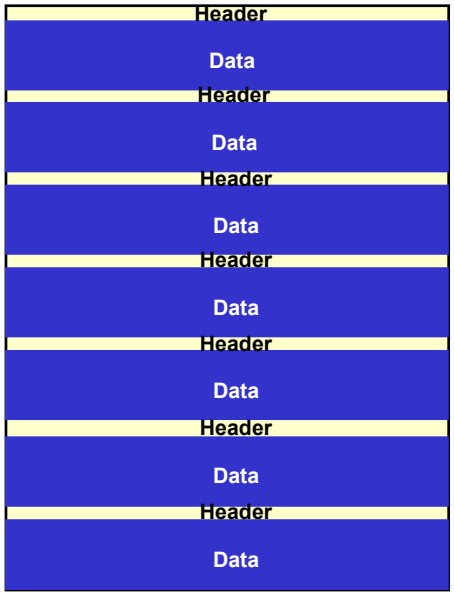
Auto-Deposit Automatically Loads Header into Cache and Data into Main Memory



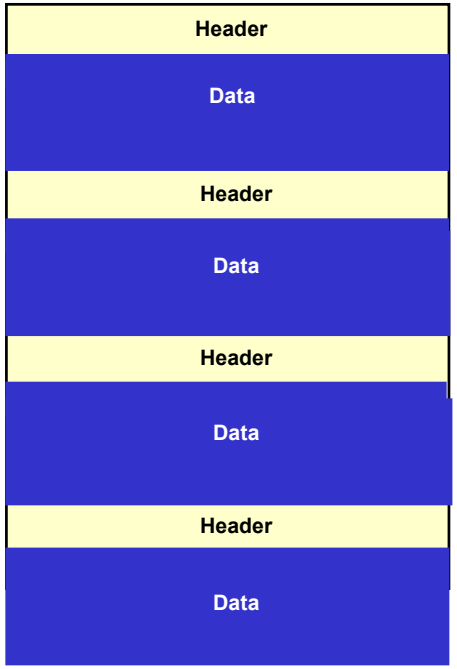
Autodeposit™ Operation Uses HyperTransport™ Registers to Write Packet Headers to L2 and Packet Data to Main Memory



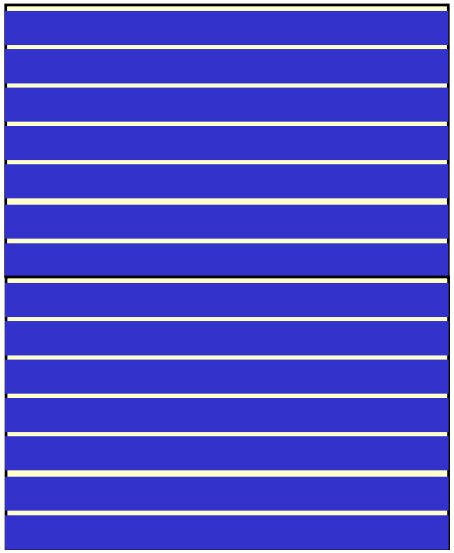
Autodeposit™ Operation can define Multiple Memory Spaces and Header/Data Sizes



Buffer Region 1

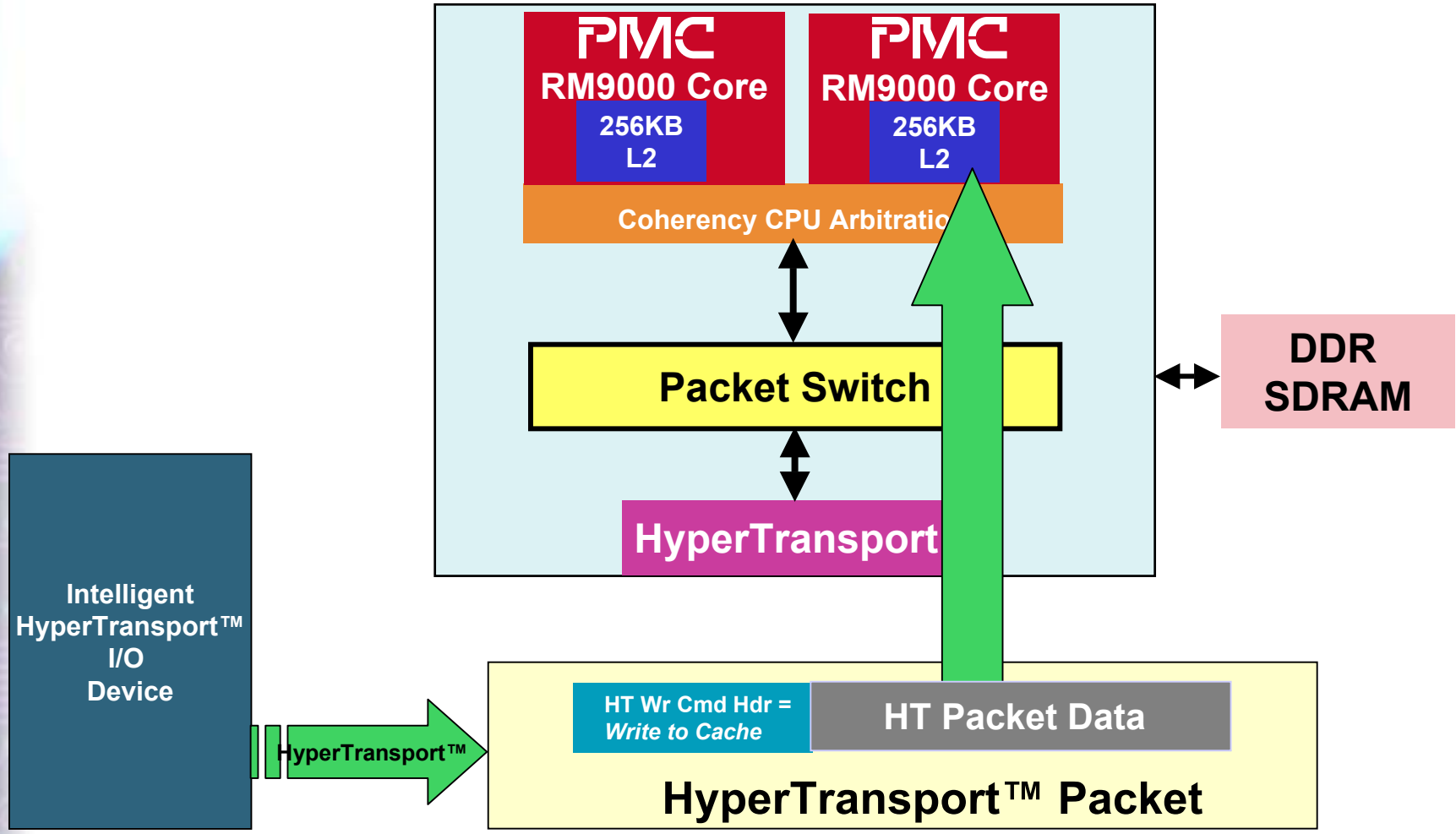


Buffer Region 2

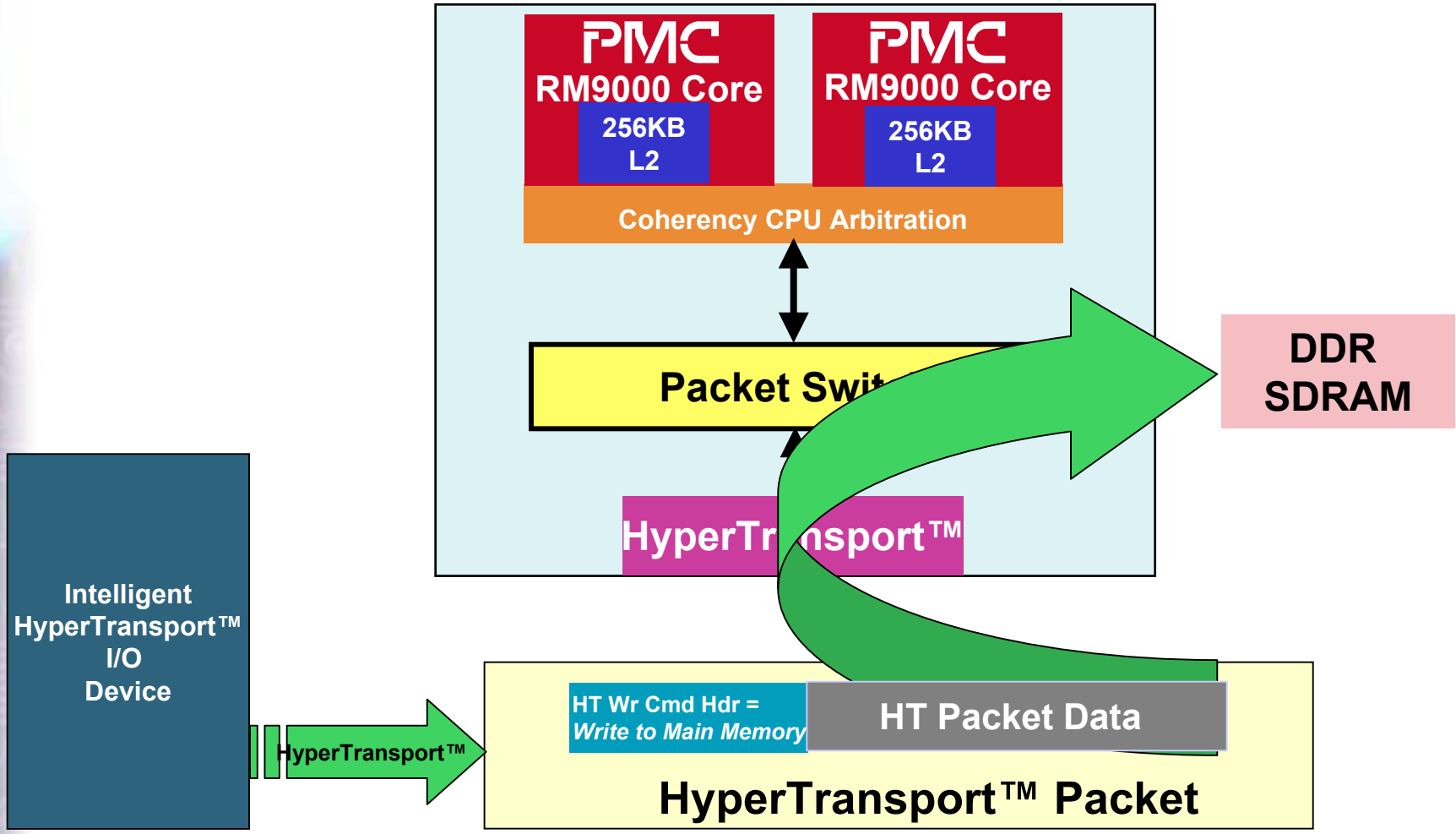


Buffer Region 3

Live-Deposit Operation allows Intelligent HyperTransport™ Peripheral to Write HyperTransport™ Packets Directly to Cache

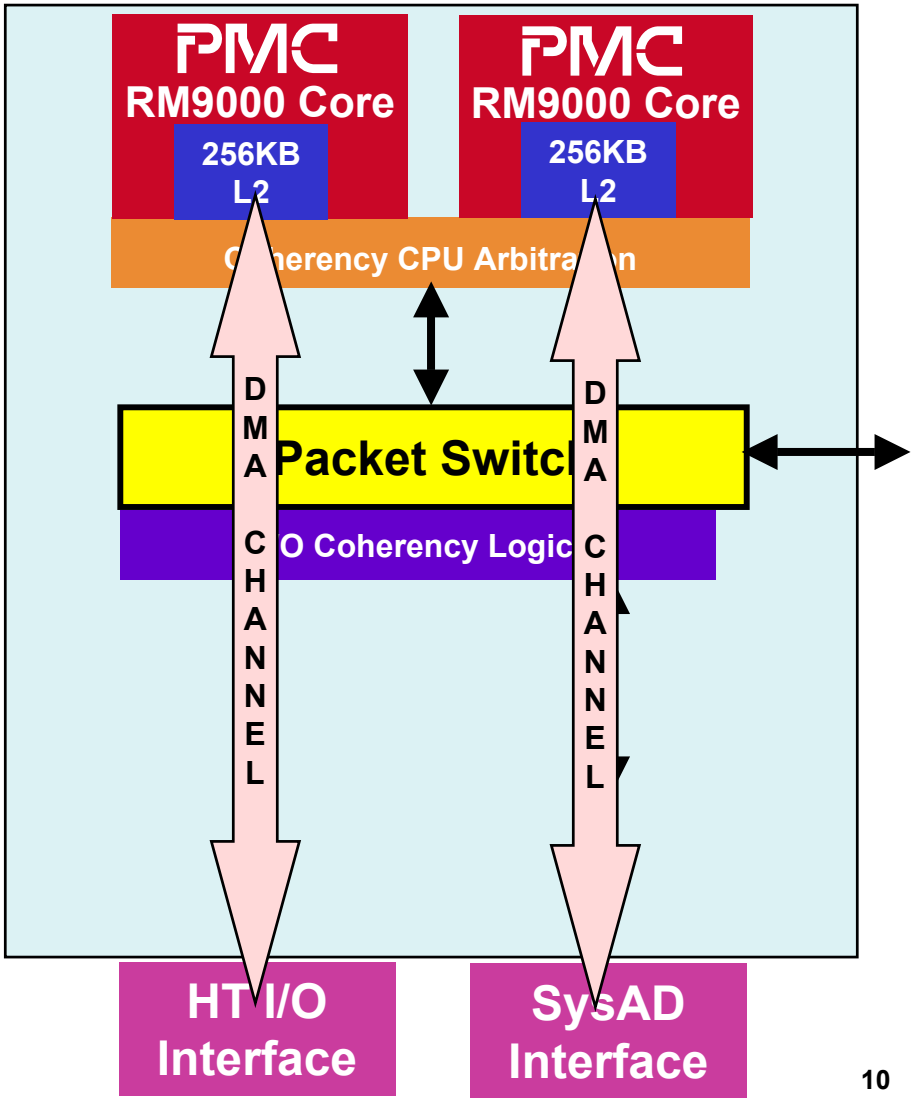


Live-Deposit Operation also Supports Writing HyperTransport™ Packets into Main Memory

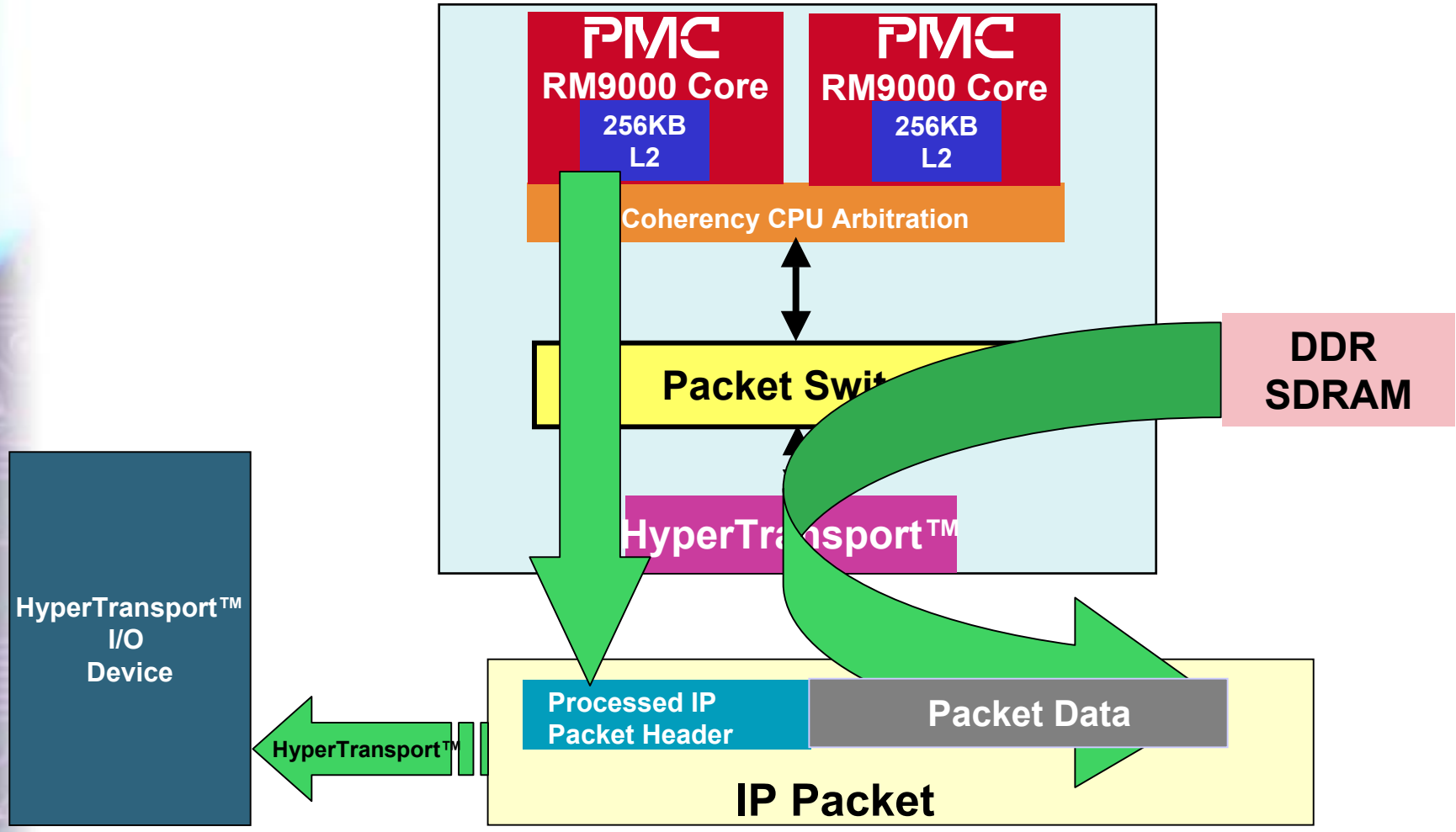


Live Deposit Operation Provides DMA Access to Cache

- Uses DMA Channel Configuration Register to Select Writing into Cache or Main Memory
- Supports Writing to L2 over both HyperTransport™ and SysAD Bus



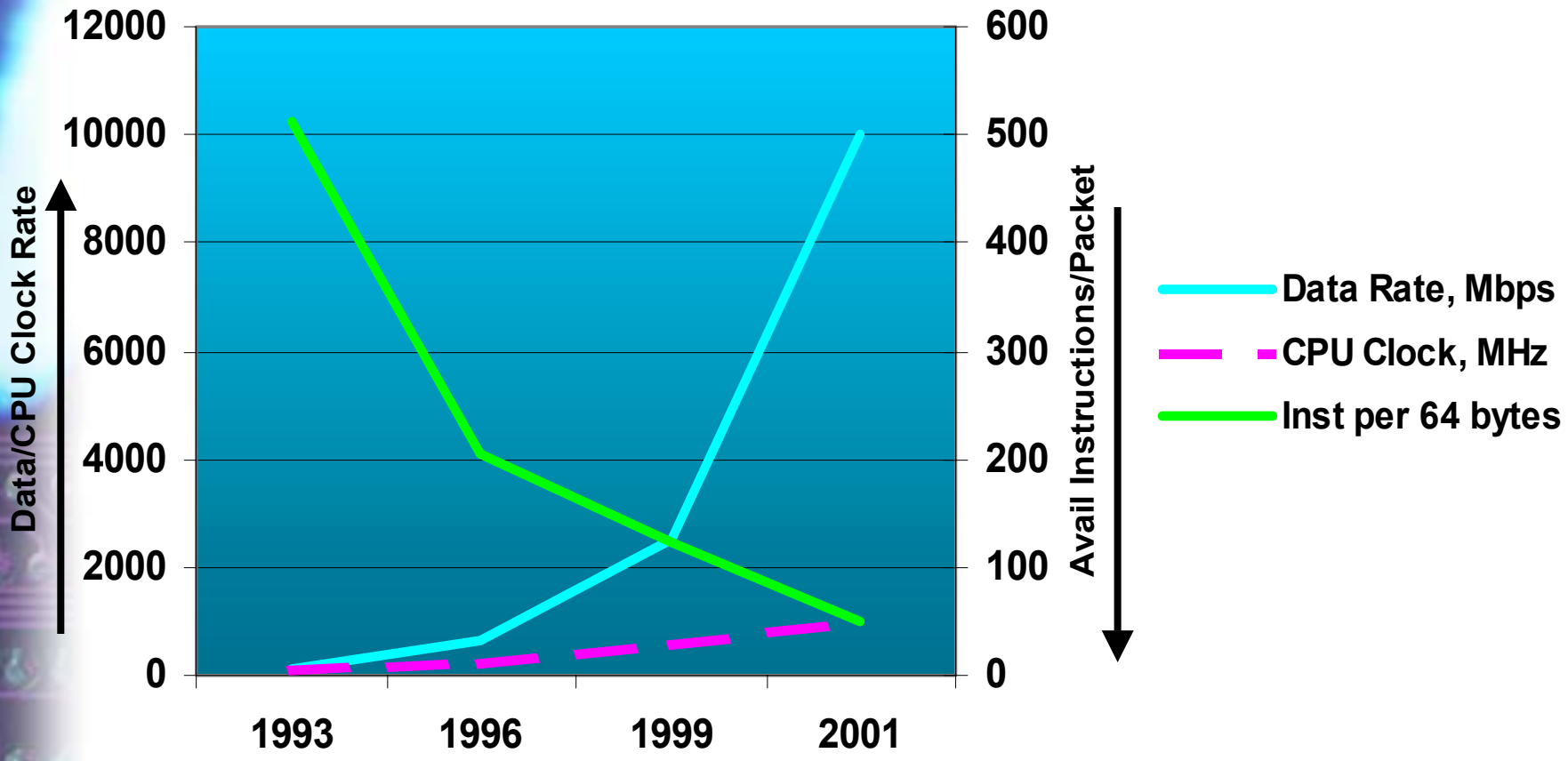
DMA Controller Efficiently Writes Modified Packet Header and Packet Data to HyperTransport™ Peripheral



CPU Subsystem Solves Problems found in Traditional Processors

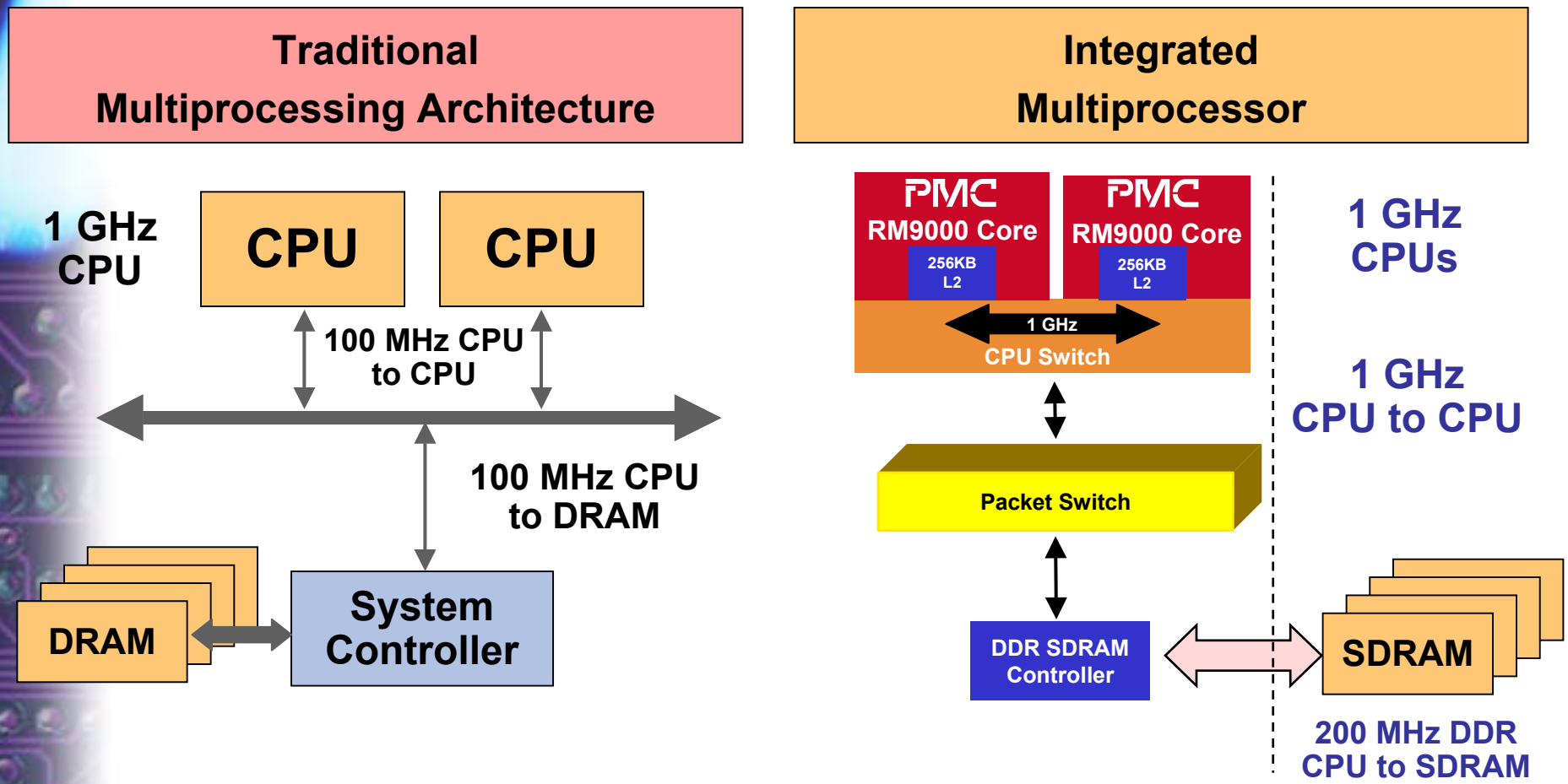
- Dual Integrated 1 GHz Processors
- Inter-CPU Data Transmission at 1 GHz CPU Frequency
- Modified Cache Lines can be Passed Directly Between CPUs
- Multiple Ways to Extract Parallelism

Why Multiple CPUs?



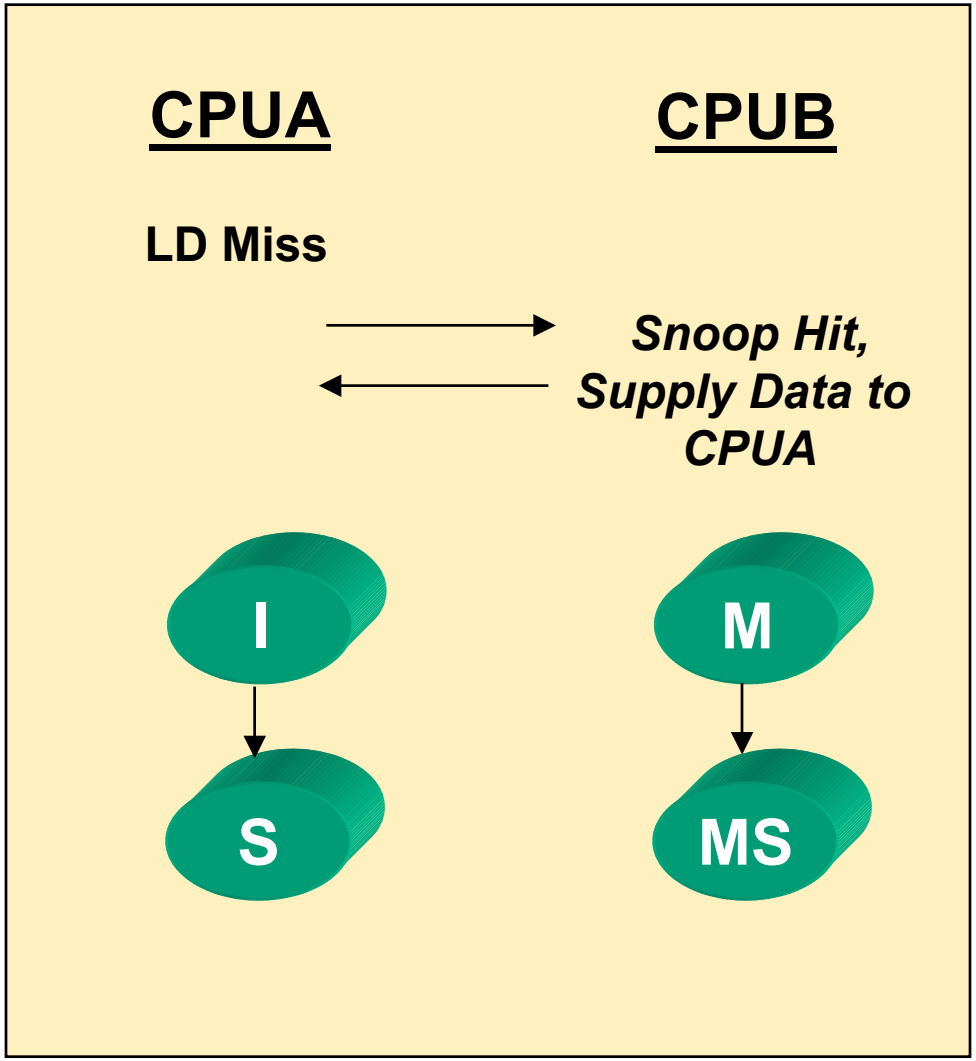
- Simply Scaling the CPU Clock is Not Enough
- Efficient Parallel Processing Supplies Geometric Scaling

Solving the Multiprocessing Bottleneck



MOESI Cache Coherency Allows Sharing of Modified Data

- **Modified-Shared State Occurs When CPUA Accesses Memory That is in CPU B's Cache in the Modified State**
  - LD Miss in CPUA
  - Snoop Hit in CPUB
    - **CPUB Gives Cache Line to CPUA**
- **Minimizes Accesses to Main Memory**
  - Avoids 3 Party Transactions
  - Minimizes Latency



Extracting Parallelism

Packet Pipeline	Packet Analysis →	Packet Editing
Parallel Packets	Packet 1	Packet 2
Dedicated Tasks	Operating System	Packet Processing
Fully Symmetric	SMP OS → Tasks	Tasks

Summary

- **Packet Processing Requires Very High Performance in Both the I/O and CPU Subsystems**
- **RM9000x2 I/O Subsystem**
  - 16 Gbit/s HyperTransport™ Interface
  - Direct Deposit Cache™ Permits I/O Devices Direct Access into Cache Memory
- **RM9000x2 CPU Subsystem**
  - Dual GHz CPU Cores
  - Cache Coherency enforced between CPU Cores using 5-State MOESI Protocol
- Visit **PMC-Sierra's MIPS Resource Page** at http://www.pmc-sierra.com/processors